MaxStream Incorporated

Exceeding Your Wireless Expectations

9XStream™

Wireless OEM Module

Operating Manual v2.8

MaxStream, Inc.
1215 S. 1680 W.
Orem, UT 84058
Phone: (801) 765-9885
Fax: (801) 765-9895
info@maxstream.net
http://www.maxstream.net

TABLE OF CONTENTS

MaxStream Inc.
1215 W. 1680 W.
Orem, UT 84058
(801) 765-9885

9XStream Frequency Hopping Data Module

FCC Compliance Warning:

Changes or modifications to the 9XStream Data Module not expressly approve d by MaxStream, Inc. could void the user's authority to operate this product.

Note: This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

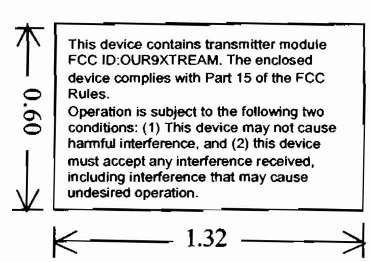
This device complies with part 15 of the FCC Rules. Operation is subject to the following two conditions:

(1) This device may not cause harmful interference, and

(2) This device must accept any interference received, including interference that may cause undesired operation.

External Label

All OEM products must have this label placed in a visible location. MaxStream module s ship with such labels on the shield. If the module is hidden, a separate label must be attached to the product in a visible location.



General Description

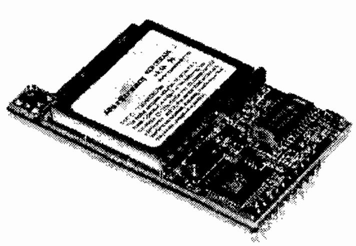
The 9XSTREAM-192/96/12 modules are 100-milliwatt frequency hopping wireless modules that allow wireless communication between equipment using a standard asynchronous serial data stream. The module is half-duplex and can sustain a continuous data stream at the specified data rate. Boasting excellent sensitivity and range, the 9XSTREAM module is perfect for applications such as Supervisory Control and Data Acquisition (SCADA), remote meter reading, home automation, security, instrument monitoring, point of sale systems (POS), and countless other applications. The 9XSTREAM operates within the 900 MHz ISM Band under Part 15 of the FCC Rules and Regulations and is FCC approved. A regulated 5-volt supply is required for operation.

Features

- Frequency Hopping Spread Spectrum (FHSS) technology
- Noise and interference resistance
- Excellent sensitivity and range
- Several low power modes down to 1 microamp
- Standard serial digital interface
- Networking and addressing
- Simple AT command interface

Simple Product Integration

- Easy to integrate; no knowledge of RF required
- Interfaces to any microcontroller's UART or PC Serial Port (using the MaxStream interface board)
- Small size
- Exceptional data transfer performance
- FCC approved, no further licensing or approval necessary



Block Diagram

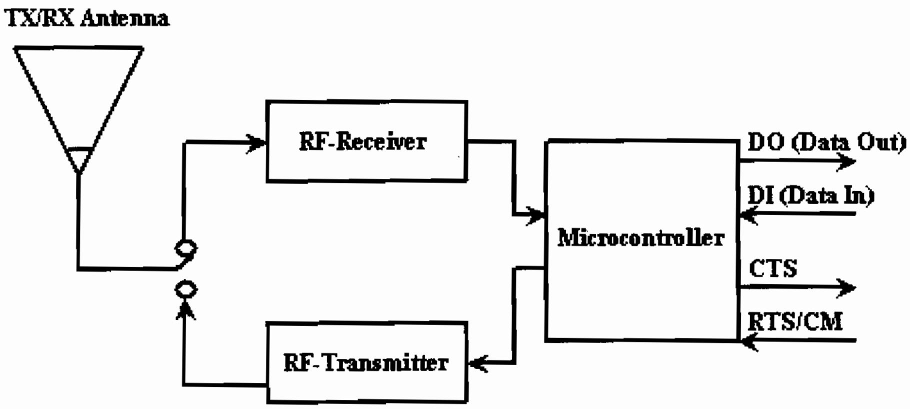


Figure 1 – Block diagram demonstrating basic module operation and data flow for both transmit and receive.

Diagram

The 9XStream data module connects to a host device using an 11 pin header and a 4 pin header (0.1" spaced).

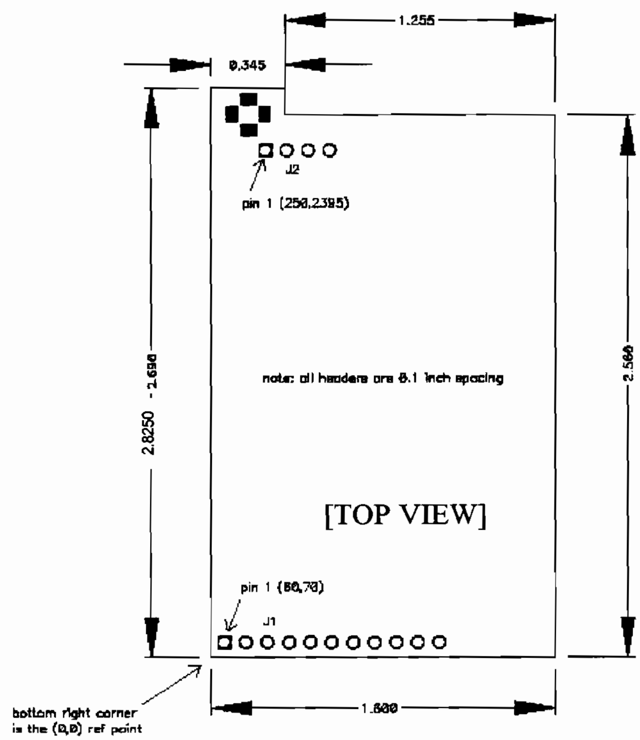


Figure 2 – Top view diagram of the 9XStream module with pin layout and dimensions.

J1 Pin Descriptions

Pin No.	Pin Name	I/O Type	Description	Active
1	$\overline{\text{CTS}}$	O PU	Clear to Send flow control	low
2	SLEEP (PWRDN)	I PU	Can be used to enter Sleep Mode (See "Modes of Operation" section for details.)	high
3	DO (Data Out)	O PU	Data leaving the module that is sent to the host	high
4	DI (Data In)	I	Data entering the 9XStream module to be transmitted over the air	high
5	RTS/CMD	I PD	Command mode enable (See "Binary Command Mode" section for details.)	high
6	$\overline{\text{RESET}}$	I PU	Reset module	low
7	RXLED	O	Indicates good RF data reception	high
8	$\overline{\text{TX}}$/PWR	O	PWR - Indicates module powered on	high
			$\overline{\text{TX}}$ - Asserted during transmission	? low
9	$\overline{\text{CONFIG}}$	I PU*	Hold low during power up or reset - forces ASCII command mode. DO NOT TIE TO MICROPROCESSOR! (See "Serial Port Operation" section for details.)	low
10	VCC	I	+5 VDC	-
11	GND	-	Signal ground	-

PU – 10kO Pull-Up Resistor
PD – 10kO Pull-Down Resistor

* $\overline{\text{CONFIG}}$ signal has a 47kO Pull-Up Resistor

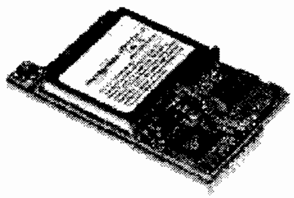
J2 Pin Descriptions

Pin	Signal
1	GND
2	GND
3	GND
4	GND

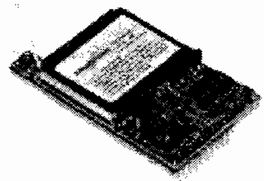
Specifications

	X09-009	X09-019
General		
Frequency Range	902 to 928 MHz, unlicensed ISM Band	
Type	Frequency Hopping Spread Spectrum Transceiver	
Frequency Control	Direct FM	
Transport Protocol	Various Monitoring and Addressing Modes – *see "Networking and Addressing" section*	
Channel Capacity	Hops through 25 channels. Features 7 different hop sequences.	
Serial Data Interface	Asynchronous CMOS (TTL) signals, 5V levels	
Serial Interface Baud Rate	Configurable from 2400-57600 bps **1200 bps available*	Configurable from 2400-57600 bps
Data Throughput	9600 bps	19200 bps
Performance		
Channel Data Rate	10k	20k
Transmit Power Output	100mW	100mW
Rx Sensitivity	-110 dBm	–107 dBm
Range*	Indoor: 600' to 1500' Outdoor: 7mi. with dipole, over 20 mi. with high gain antenna	Indoor: 425'to 1060' Outdoor: 5 miles with dipole, over 14 miles with high gain antenna
Interference Rejection	70 dB at pager and cellular phone frequencies	
Power Requirements		
Supply Voltage	5 VDC +/-0.25V	
Current Consumption	Tx – 150 mA nominal, Rx – 50 mA nominal Power Down mode – less than 1 microamp Other cyclic power-down modes available – *see "Low Power Modes" section*	
Physical Properties		
Board Size	1.6" x 2.85" x .35" (4.06 x 6.86 x .89) cm	
Weight	0.8oz (24g)	
Connectors	11 pin and 4 pin 0.1" spaced male Berg type headers	
Operating Temperature	Standard: 0°C to 70°C Industrial version: -40°C to 85°C available	
Antennas		
Antenna Connector	MMCX Female or Reverse Polarity SMA Male	
Approved Antennas	Integral wire antenna (factory installed) 1/4 wave flexible monopole 1/2 wave flexible dipole, SMA	

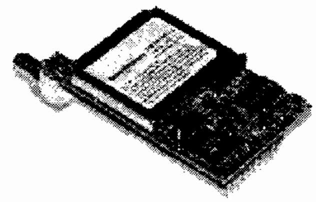
Product Listing



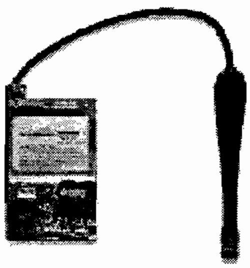
MMCX – No Antenna
X09-009NM, X09-019NM



MMCX – Wire Antenna
X09-009WM, X09-019WM

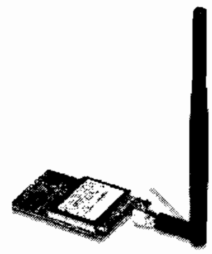


SMA – No Antenna
X09-009NS, X09-019NS



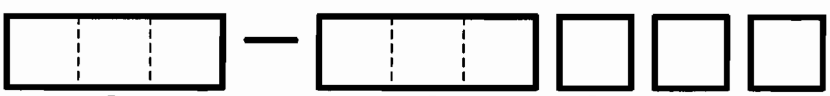
¼ Wave Antenna MMCX
A09-QBMM-3-P6I

½ Wave Antenna MMCX
A09-HBMM-7-P6I



½ Wave Antenna SMA
A09-HASM-675

Module Part Numbers



① Operating Frequency
- X09 – 900 MHz
- X24 – 2.4 GHz

② Over-Air Baud Rate
- 009 – 9600-baud
- 019 – 19200-baud

③ Wire Antenna
- W – Wire Antenna
- N – No Wire Antenna

④ Connector
- M – MMCX Connector
- S – SMA Connector

⑤ Temperature Rating
- C – Commercial. Certified for operation from 0° to 70° C.
- I – Industrial. Conformal coated. Certified for -40° to 85° C.
- T – 100% Tested. Conformal coated. 100% tested -40° to 85° C.

900 MHz Modules

Product Code	Description
X09-009NM	9600-baud, no wire mount antenna, MMCX connector
X09-009WM	9600-baud, wire mount antenna, MMCX connector
X09-009NS	9600-baud, no wire mount antenna, SMA connector
X09-009WS	9600-baud, wire mount antenna, SMA connector
X09-019NM	19200-baud, no wire mount antenna, MMCX connector
X09-019WM	19200-baud, wire mount antenna, MMCX connector
X09-019NS	19200-baud, no wire mount antenna, SMA connector
X09-019WS	19200-baud, wire mount antenna, SMA connector
Package Kits	
X09-009PK	9600-baud, Package
X09-019PK	19200-baud, Package
Development Kits	
X09-009DK	9600-baud, Development Kit
X09-019DK	19200-baud, Development Kit

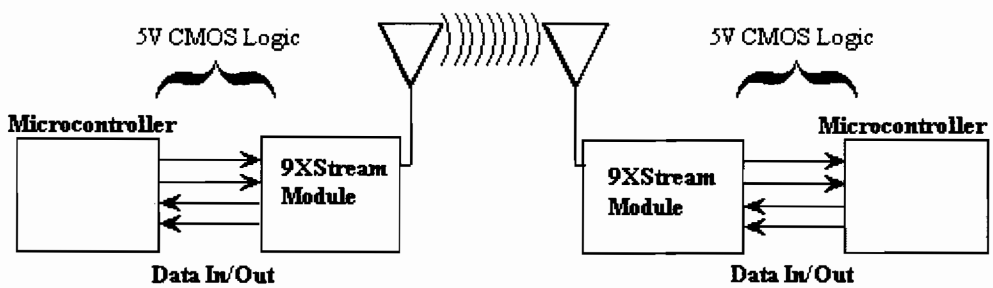
Antennas

Product Code	Gain (dBd)	Frequency Range (MHz)	Feed Connector	Length	Description
Yagi Antennas					
A09-Y4NF	6	896 – 980	N Female	14"	4-element Yagi
A09-Y6NF	8.5	902 – 928	N Female	25"	6-element Yagi
A09-Y7NF	10	902 – 928	N Female	31"	7-element Yagi
A09-Y10NF	12	902 – 928	N Female	42"	10-element Yagi
A09-Y12NF	11	902 – 928	N Female	50"	12-element Yagi
A09-Y15NF	15	902 – 928	N Female	60"	15-element Yagi
Base Station Antennas					
A09-W5NF	5	902 – 928	N Female	23"	Fiberglass Base Station
A09-F3NF	3	902 – 928	N Female	25"	Fiberglass Base Station
A09-F6NF	6	902 – 928	N Female	65"	Fiberglass Base Station
Whip Antennas					
A09-HASM-675	0	902 – 928	RPSMA	7"	½ wave Whip
A09-HBMM-7-P6I	0	902 – 928	MMCX	7"	½ wave Whip
A09-QBMM-3-P6I	0	902 – 928	MMCX	3"	¼ wave Whip
NOTE: If using an external antenna, the wire antenna should be removed from the 9XStream module.					

Serial Port Operation

The 9XStream modules come equipped with a CMOS-level asynchronous serial port. Through this serial port, the 9XStream can communicate directly with any device having a UART interface, or with a PC, or other RS-232 port, via the MaxStream interface board. By connecting the 9XStream to a host device's serial port, the host device becomes empowered to communicate wirelessly with ease. To transmit, the host device simply sends serial data to the 9XStream and the 9XStream converts the data to spread spectrum FCC-approved wireless data. When this spread spectrum data is detected by the receiving 9XStream module, the data is checked for integrity and then sent to a receiving device via the serial port. This is shown in Figure 3 below.

a.



b.

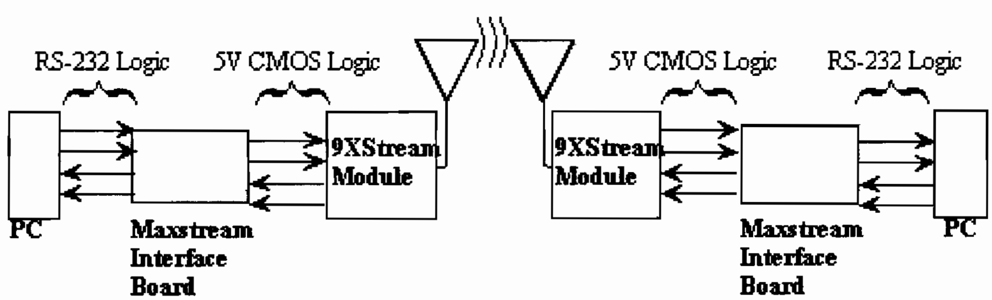


Figure 3 – Diagram of data flow from a microcontroller or PC (or any RS-232 device) through a pair of 9XStream wireless modules to receiving microcontroller or PC. (Note: When connecting the 9XStream to an RS-232 device, the MaxStream Interface Board adjusts voltage levels between the RS-232 device and the MaxStream wireless module.)

Serial Pins

Figure 3 above shows 4 data lines needed to interface from a microcontroller or RS-232 device to the 9XStream modules. These four lines represent DI (Data In), DO (Data Out), *CTS, and RTS/CMD (request to send/command mode). (All low-asserted pins are distinguished with a line over the top of the pin name, or a '*' symbol prefacing the pin name.) While the DI and DO pins are indispensable in almost all cases, the *CTS and RTS/CMD may not be needed under certain conditions. The following includes a brief description of each of these pins and under what conditions the pins must be used. A brief explanation of the *CONFIG pin is also provided.

DI (Data In) – Pin 4 (Input)

Data enters the 9XStream the DI pin as an asynchronous serial signal. The serial signal is idle (high) when no data is being transmitted. Each data packet consists of a start bit (low), 8 data bits, and a stop bit (high) as shown below in Figure 4.

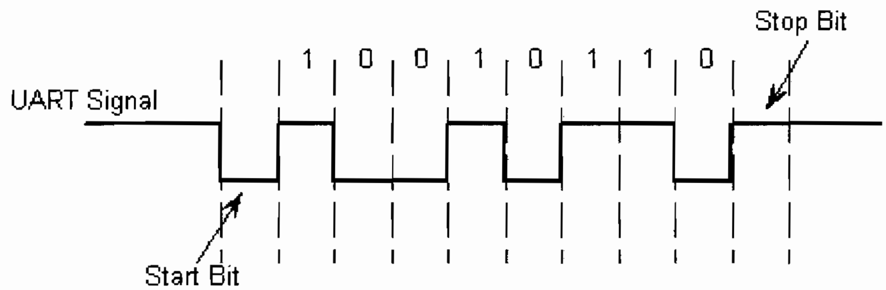


Figure 4 – UART signal transmitted to 9XStream module from microcontroller or RS-232 device.

The 9XStream modules transfer exactly 8-bits over the air. The start and stop bits from the UART signal are not actually transmitted, but are regenerated on the receiving module before they are sent out the serial port. This allows for the following data configurations to be sent:

- 8-bit, no parity, 1 stop bit
- 7-bit, even parity, 1 stop bit
- 7-bit, odd parity, 1 stop bit
- 7-bit, no parity, 2 stop bits

Since the 9XStream is half-duplex, it can only transmit or receive at a given time. Thus, once the first byte is detected on the DI pin, the 9XStream immediately begins transmitting the incoming data unless over-the-air data is already being received. In this case, the data on the DI pin is stored in the data buffer until data is no longer being received at the antenna. If the 9XStream receives a lengthy sequence of data, the data buffer could reach its capacity (132 bytes) in which case the *CTS signal will need to be implemented (see *CTS section below).

Also, the 9600 and 19200-baud modules allow the incoming serial data to be transferred at a rate of 2400-57600 bits/second. Serial data can be transferred to the module at a rate equal to or less than the module's over-the-air baud rate without any problems. However, if the serial interface rate is set to exceed the module's baud rate (9600 or 19200 bps respectively), *CTS must be implemented since the data buffer may become full.

DO (Data Out) – Pin 3 (Output)

Data received from over-the-air transmissions is checked for errors and then sent to the DO pin.

$\overline{\text{CTS}}$ – Pin 1 (Output)

The $\overline{\text{CTS}}$ pin (clear to send) informs the host device whether or not serial data can be sent to the 9XStream module. When *CTS is low, serial data can be sent to the 9XStream module. All incoming serial data is stored in a data buffer until the next data packet is transmitted (over-the-air). The data buffer can hold up to 132 bytes of data. Once the data buffer contains 115 bytes, the 9XStream module drives the *CTS signal high (de-asserts it) to alert the host device to stop sending serial data. *CTS remains high until less than 98 bytes are in the buffer.

There are three cases in which the data buffer may become full.

1) Since the 9600 and 19200-baud modules support serial interface rates from 2400-57600 bits/second, the serial data rate could be configured at a higher rate than the module's over-the-air baud rate. If this happens, long serial data streams can cause the data buffer to become momentarily full, causing *CTS to de-assert.

2) Since the 9XStream modules are half-duplex, they can either receive or transmit at a given time. If a long string of over-the-air data is received, serial data could arrive at the buffer and cause it to become full if the module is unable to transmit the data.

3) If any module in a network (see "Networking and Addressing") is transmitting data, all other modules in the network will not transmit until they finish receiving data. Thus, if they receive serial data, their data buffers could become full.

In some applications, where none of these conditions will occur, the *CTS signal need not be monitored.

RTS/CMD – Pin 5 (Input)

RTS
The RTS signal (request to send) is not implemented for flow control with the 9XStream modules. All received data (over-the-air) is sent out the serial port regardless of the RTS signal.

CMD
The 9XStream comes with a variety of configurable settings including power-saving modes and network addressing options. This pin can be used as one way to manually configure the 9XStream module as described in the "Command Mode" section. When this pin is driven high (asserted), incoming serial data (on the DI pin) is interpreted as commands instead of data. See the "Command Mode" section for more information.

***CONFIG – Pin 9 (Input)**

The (low-asserted) *CONFIG pin is used to force the module to enter AT Command Mode. When asserted, the serial port baud rate is temporarily set to match the default baud rate of the 9XStream module. This assures that the module will transition into AT Command Mode at a known baud rate. Upon entering AT Command Mode, all configured parameters, including the baud rate, remain in their saved state and can be modified as described in the "AT Command Mode" section.

NOTE: DO NOT tie the *CONFIG pin to an external device as it may cause problems with module operation. The *CONFIG pin should be tied to an external switch and used manually to enter AT Command Mode only when AT Command Mode cannot be entered under the normal procedure (see "AT Command Mode" section).

Modes of Operation

The 9XStream wireless module features several modes of operation that allow the module to be responsive to data and yet utilize minimum power. The figure below shows these modes and is followed by a comprehensive look into each mode and the necessary conditions for the 9XStream module to transition from one mode to another.

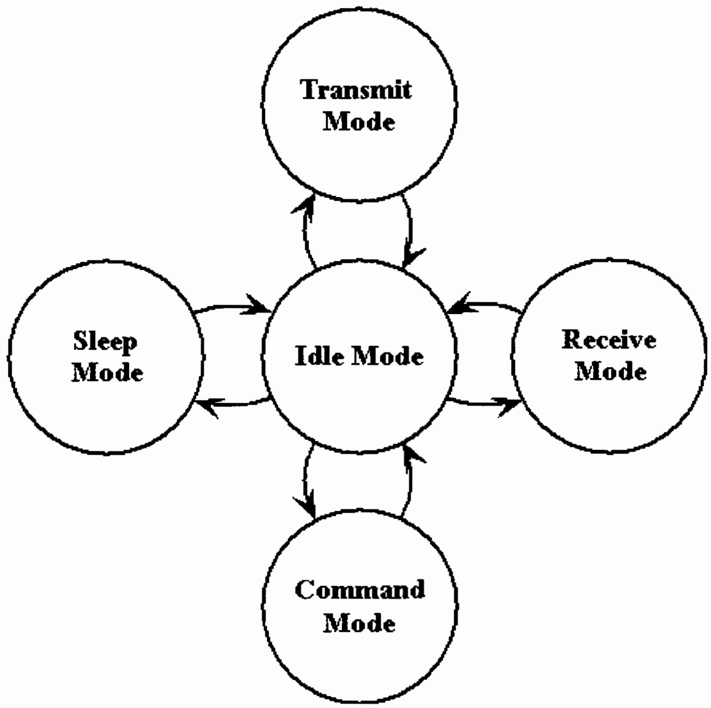


Figure 4 – State diagram demonstrating modes of operation for the 9XStream module.

Idle Mode

The 9XStream module operates in Idle Mode when there is no data being transmitted or received. The module will transition to Transmit Mode once data is presented on the DI pin. If valid data is detected at the antenna, the module will switch from Idle Mode to Receive Mode. When no longer transmitting or receiving, the module returns to Idle Mode.

Transmit Mode

When the first byte arrives in the data buffer through the DI pin, the module leaves Idle Mode and transitions to Transmit Mode. This transition happens instantaneously from the moment the first byte of data arrives in the data buffer. In Transmit Mode, a header is sent out and is then followed by the first data packet, which has a CRC (cyclic redundancy check) attached (see "Data Validity" section below for more information). The first data packet contains all bytes that accumulated in the data buffer while the header was being sent. After the first data packet is sent, if more data is available in the buffer, another header will be sent, followed by another data packet. This second data packet (and all subsequent data packets) will consist of data that accumulated in the buffer while the previous data packet and header were being sent out (see Figure 5b below). The size of each data packet can vary up to 64 bytes. This progression is shown in Figure 5a.

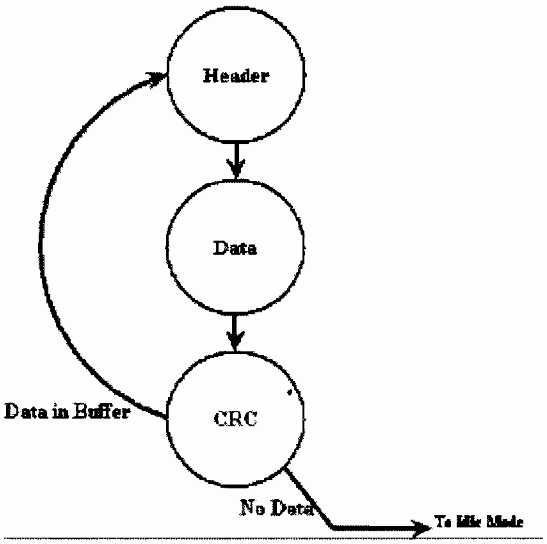


Figure 5a – Transmit Mode description.

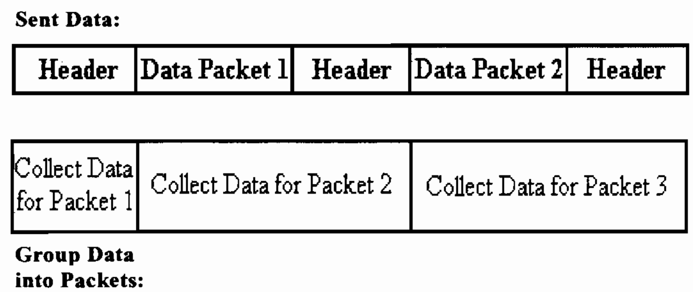


Figure 5b – Generation of data packets.

Data Validity

To verify data integrity, a 16-bit cyclic redundancy check (CRC) is computed for the transmitted data and attached to the end of each data packet before transmission. The receiver will then compute the CRC on all incoming data. Any received data that has an invalid CRC is discarded.

Transmission Latency

The time required to send a packet of data depends on the number of bytes being sent, and the baud rate. In addition, the modules have a synchronization timer option that can be manually configured using the SY command as discussed in the "9XStream Commands" section. Modifying this parameter can significantly change the transmission latency. See the "Timing Diagrams" section for more information on transmission latencies.

A Note About Headers

As noted in Figure 5, data packets are always prefaced by a header. The header contains information that is used by all receivers (within range) to synchronize their hopping patterns to the transmitter. The length of the header can be reduced in some applications by eliminating the synchronization information. See "Timing Diagrams" for more information.

Receive Mode

If over-the-air data is present at the RF receiver when the module is in Idle Mode, it will transition to Receive Mode and start receiving packets. Once a packet is received, it goes through a CRC (cyclic redundancy check) to ensure that the data was transmitted correctly. If the CRC data bits on the incoming packet are invalid, the packet is discarded. If the CRC is valid, the packet is sent to the serial port via the DO pin. This process is shown in Figure 6 below.

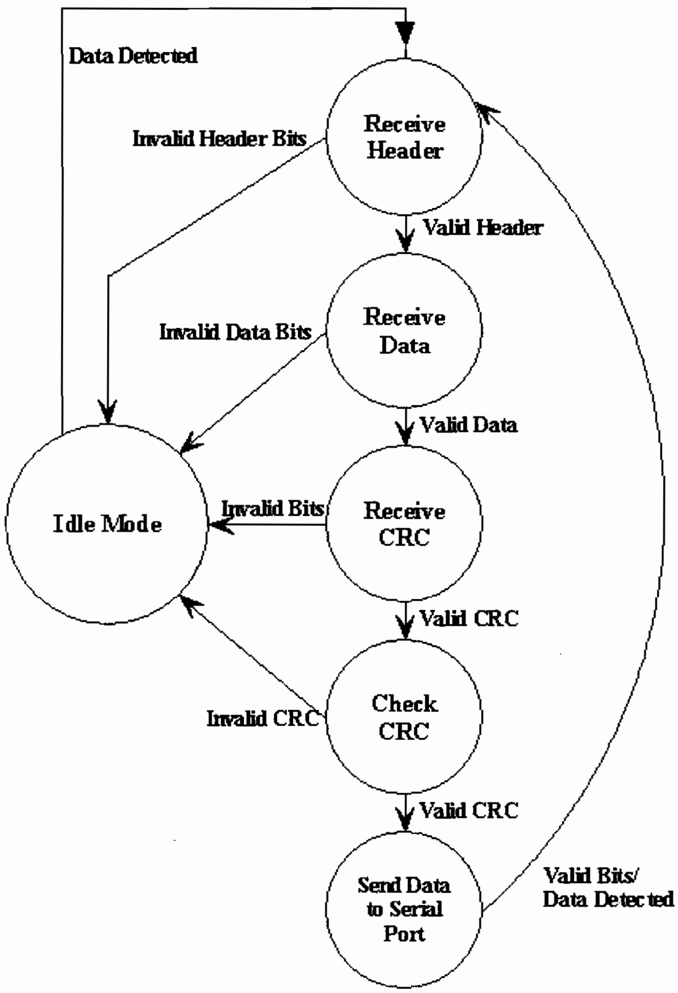


Figure 6 – Receive Mode description.

The module will remain in Receive Mode until an error is detected in the received data, or no more data is detected, at which point, the module transitions to Idle Mode. If serial data was stored in the data buffer while the module was in Receive Mode, the data will be transmitted after the module returns to Idle Mode.

Sleep Mode

Sleep Mode enables the 9XStream module to go into a low-power state where minimal power is consumed when the module is not in use. Once in Sleep Mode, the module will not transmit or receive data until it first returns to Idle Mode. To enter Sleep Mode, the module must be inactive (no data transmission or reception) for a user-defined period of time (specified by the ST command). After this time elapses, the module transitions to Sleep Mode. By default, Sleep Mode is disabled and must be enabled using the SM command.

The 9XStream features several Sleep Mode settings, each of which makes use of different mechanisms to enter or leave Sleep Mode. The following table lists the different Sleep Mode settings and the requirements to transition to and from Sleep Mode for each setting.

Sleep Settings	**Transition to Sleep Mode**	**Return to Idle Mode**	**Sleep Current**		**Configure these Commands**
No Sleep Mode	None. The module remains in Idle Mode. (Default Setting)	-	50 mA		-
Pin Sleep	High on SLEEP pin (pin 2).	Low on SLEEP pin.	<2 uA		SM
Serial Port Sleep	Automatic transition after a user-defined period of module inactivity (no transmitting or receiving of data). Set by ST command.	Any serial byte received on DI pin.	1 mA		SM, ST
Cyclic Sleep	Automatic transition after a user-defined period of module inactivity (no transmitting or receiving of data). Set by ST command.	Periodically returns to Idle Mode	**0.5 Cyclic:** Min: 60μA Max: 40mA Avg: 7.4mA **2.0 Cyclic:** Min: 60μA Max: 35.7mA Avg: 2.07mA **8.0 Cyclic:** Min: 60μA Max: 35.56mA Avg: 630 μA	**1.0 Cyclic:** Min: 60μA Max: 35.7mA Avg: 4mA **4.0 Cyclic:** Min: 60μA Max: 35.7mA Avg: 1.1mA	SM, ST, LH, HT

The following sections describe how each setting operates.

Pin Sleep (SM=1)

After enabling the Pin Sleep setting, the SLEEP pin (Pin 2) controls whether the 9XStream is active or in Sleep Mode. If SLEEP is low, the module is fully operational. Once SLEEP is driven high (asserted), the module transitions to Sleep Mode and remains in its lowest power-consuming state until the SLEEP pin is de-asserted. The 9XStream requires 85 ms to transition from Sleep Mode to Idle Mode. The SLEEP pin is only active if the module is set up to operate in this mode; otherwise the pin is ignored. (See the 'SM' command in the "9XStream Commands" section for more information.) Once in Pin Sleep Mode, the *CTS pin (Pin 1) is driven high (de-asserted) to indicate that data should not be sent to the module during this time. The TX/PWR pin (Pin 8) is driven low (de-asserted) when the module is in Pin Sleep Mode.

Serial Port Sleep (SM=2)

If this state is enabled, the module goes into Sleep Mode after a user-defined period of inactivity (no transmitting or receiving of data). This period of time can be changed by modifying the ST command. When the module is in Serial Port Sleep Mode, the TX/PWR pin (Pin 8) is de-asserted. The module will return to Idle Mode once a character is received on the DI pin.

Cyclic Sleep (SM=3-7)

If the Cyclic Sleep setting is enabled, the 9XStream module goes into Sleep Mode after a user-defined period of inactivity (no transmission or reception on the RF channel). This user-defined

period can be set by adjusting the ST parameter (see the 'ST' command in "9XStream Commands" section). The module remains in Sleep Mode for a user-defined period of time ranging from 0.5 seconds to 8 seconds (adjustable using 'SM' command as described in "9XStream Commands" section). After this period of time, the module returns to Idle Mode and listens for a valid data packet. If no valid data packet is found on any channel, the module returns to Sleep Mode. If a data packet is found, the module transitions into Receive Mode and receives the incoming packets until another ST inactivity time out. When the module is awake, it requires 100 milliseconds to search for a valid data packet.

While the module is in its low-power state, the *CTS pin (Pin 1) is driven high (de-asserted) to indicate that data should not be sent to the module during this time. When the module awakens to listen for data, the *CTS pin is asserted, and any data received on the DI pin will be transmitted. The TX/PWR pin (Pin 8) is driven low (de-asserted) when the module is in Cyclic Sleep Mode. It is asserted each time the module cycles into Idle Mode to listen for valid data packets, and then de-asserts if the module returns to Sleep Mode.

A Note About Cyclic Scanning

Each RF packet consists of a header and data as shown previously in Figure 5b. Since the header contains the channel synchronization information, the module must wake up during the header portion of a packet in order to synchronize with the transmitter and receive the data. To ensure that the 9XStream module can detect the header, a long header can be sent periodically during a transmission. This long header repeats the synchronization information for a period of time defined by the 'LH' command.

By default the long header is turned off, and must be enabled in order to communicate with a module operating in Cyclic Sleep Mode. To enable the long header, the LH parameter must be set to a value greater than the time of cyclic sleep to ensure accurate detection by the receiver(s). For example, if the 9XStream is set to wake up from Sleep Mode every four seconds and check for a packet, a transmitter would need to send a long header that is just over four seconds in length to guarantee that the receiving module will detect the packet. (The exact timing requirements can be found in the Timing Diagrams section.) This concept of long header length versus Sleep Mode timing is displayed in Figure 7 below.

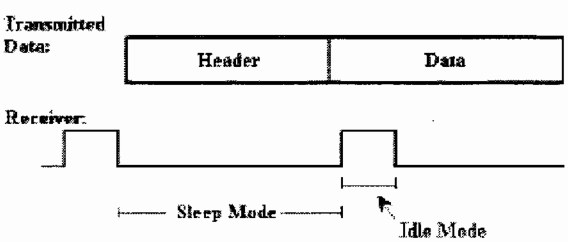


Figure 7a – The length of the long header is not as long as the period of Cyclic Sleep. It is possible for the receiver to wake and miss the header (and the data packet) in this scenario.

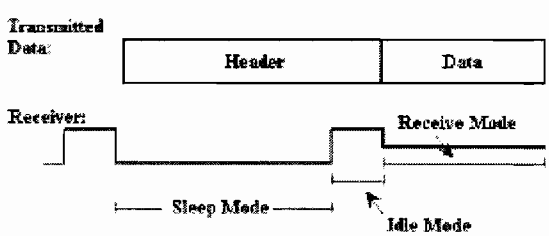


Figure 7b – The length of the long header exceeds the period of Cyclic Sleep. The receiver is guaranteed to detect and receive the data packet.

The long header is only sent with the initial transmitted packet after a user-defined period of inactivity (no serial data received and no over-the-air transmitting or receiving). This period of inactivity must be adjusted using the 'HT' command as described in the "9XStream Commands" section. Sending the long header assures that the receiver will detect the new transmission and will be able to receive the data as long as the header length slightly exceeds the cyclic sleep time.

Command Mode

Command Mode allows several features, including the power-down and addressing options, to be configured. These adjustable parameters offer greater flexibility to designers in configuring the module to fit specific design criteria. There are three ways to enter Command Mode:

1) Assert RTS/CMD and send a binary command.
2) Send the sequence "+++" to send AT commands.
3) Assert (low) the *CONFIG pin and turn the power switch off and back on. (Do not tie *CONFIG pin to microprocessor. See "Serial Port Operation" section for details.)

Once in Command Mode, the configurable parameters can be adjusted using either AT commands or Binary commands, as explained below. Any parameters that are changed while in Command Mode must be saved to non-volatile memory using the WR command, or else they will reset to their stored value upon reset or power-up.

The following sections contain a description of the AT and Binary Command Modes along with some examples. In these examples, sent or received ASCII characters are marked in quotations. The quotation marks should not be included when sending commands to the 9XStream modules. Carriage Returns (ASCII character 13) will be denoted as <CR>. Also, Binary values are represented in this document with "<" and ">" characters and are not sent as part of the actual command. All binary values are represented as hexadecimal values (HEX) in these examples, denoted by an "H" after the number. The actual Binary Command values must all be sent in binary with the least significant byte (LSB) sent first followed by the most significant byte (MSB) if the value is larger than one byte.

AT Commands

AT commands can be sent to the module using ASCII commands and parameters. A special break sequence is used so that the module will transition into AT Command Mode. The default sequence for entering AT Command Mode is:

- No characters sent for 1 second. (Time modified by "BT" command)
- Send 3 plus characters "+++" within 1 second. (Character modified by "CC" command)
- No characters sent for 1 second. (Time modified by "AT" command)

The 9XStream module will respond by sending an "OK<CR>".

All AT commands are sent as follows:

"AT"	2-Character ASCII Command	Optional Space	Parameter (HEX)	Carriage Return

The ASCII command consists of "AT" followed by two alpha-numeric bytes, and the parameter is a number represented as ASCII hexadecimal characters (0-9, A-F). The ASCII commands and parameters are not case-sensitive. The optional space can be any non-alpha-numeric character.

After executing a recognized AT command, the module responds with an OK<CR>. If an unrecognized command or a command with a bad parameter is received, the module will respond with an ERROR<CR>. The modified value will be reset upon module power-down or reset unless the "WR" command is issued to save the parameter to non-volatile memory.

To query the current value of a particular command, send the corresponding AT command without any parameters (carriage return is still sent). The response will be the current value of that command reported as a hexadecimal number.

The following example demonstrates basic AT Command functionality in the 9XStream module.

Example:
This example will change the user-defined Module Address to 1A0D (HEX) and check the current value of the SM command. It will also write the new Module Address to non-volatile memory.

Send	Response
"+++"	"OK<CR>"
"ATDT 1A0D<CR>"	"OK<CR>"
"ATSM<CR>"	"0"
"ATWR<CR>" (write to non-volatile memory)	"OK<CR>"
"ATCN<CR>" (exit AT Command Mode)	"OK<CR>"

Exiting AT Command Mode

There are two ways to exit the AT command mode and return to Idle Mode. If no valid AT commands are received within the time specified by the AT Command Timeout parameter (CT command), the module will return to Idle Mode automatically. Alternatively, the AT command mode can be exited by sending the CN command.

Binary Commands

Binary command bytes are organized as follows:

<Command><Parameters>
– 1 byte – 2 bytes

When sending a Binary command to the 9XStream, the Command byte must be sent while the RTS/CMD pin (Pin 5) is asserted. RTS/CMD can be de-asserted 100 microseconds after the stop bit of the Command byte has been sent. It does not matter whether RTS/CMD is asserted when the Parameter bytes are sent. The command will execute when all the parameters associated with the command have been sent. If all parameters haven't been received within 0.5 seconds the module will return to Idle Mode. Note that when parameters are sent, they are always two bytes long with the least significant byte sent first. When they are read, they are 1 or 2 bytes long as indicated in the Command Table below.

Binary Command Mode allows multiple commands to be sent in sequence. When the RTS/CMD pin is asserted, all incoming serial data will be interpreted as commands. Commands can be sent in sequences of commands and their associated parameters. If RTS/CMD remains asserted, all received commands will be executed by the 9XStream module. All modified parameters must be stored in non-volatile memory by sending the WR command (08H with no parameters) before powering down or resetting the module or the changes will be lost.

Commands can be queried for their current value by sending the command logically ORed with the value 80H (hexadecimal) with RTS/CMD asserted. When this binary value is sent (with no parameters) the current value of the command will be sent back, through the DO pin.

NOTE: For the 9XStream module to recognize a Binary command, the RT command must be issued from AT Command Mode to enable binary programming. If binary programming is not enabled, the module will not recognize when the RTS/CMD pin is asserted and will therefore not recognize Binary Commands.

Example:
This example will set Sleep Mode to the Pin Sleep setting and store the new Sleep Mode value to non-volatile memory. (Again, the RT command must be issued in AT Command Mode to enable binary programming before Binary Command Mode will work.)

Assert RTS/CMD	(Enter command mode.)
Send bytes:	(Send SM1 command)
<01H>	(Command Byte - SM)
<01H>	(Least significant bit of the Parameter Bytes - 01H)
<00H>	(Most significant bit of the Parameter Bytes – 00H)
Send bytes:	(Send WR command)
<08H>	(Command Byte - WR)
De-assert RTS/CMD	

9XStream Command Table

AT Command	Binary Command #	Description	Parameters	# Bytes Returned	Factory Default
DT	0 V4.08	Set the Module Address. (Only modules with the same address can communicate.)	Address value Range: 0 – FFFFH	2	0
SM	1	Adjust Sleep Mode setting.	0 – No Sleep Mode 1 – Pin Sleep 2 – Serial Port Sleep 3 – Cyclic 0.5 second sleep 4 – Cyclic 1.0 second sleep 5 – Cyclic 2.0 second sleep 6 – Cyclic 4.0 second sleep 7 – Cyclic 8.0 second sleep	1	0
ST	2	Set time to sleep. Time of inactivity before entering Sleep Mode (This number is only valid in Cyclic and Serial Port Sleep settings).	Number of tenths of seconds. Range: 0 – FFFFH.	2	64
HT	3	Set time before long header. Time of inactivity before a long header is used.	Number of tenths of seconds. Range: 0 – FFFFH. (65535 means no long header)	2	FF FF
BT	4	Set silence time before command sequence.	Number of tenths of seconds. Range: 0 – FFFFH.	2	A
AT	5	Set silence time after command sequence	Number of tenths of seconds Range: 0 – FFFFH.	2	A
CT	6	Set time out from AT command mode. Returns to Idle Mode from AT mode if no valid commands have been received within this time period.	Number of tenths of seconds. Range: 0 – FFFFH.	2	C8
FL	7	Set serial software flow control. (Hardware flow control (*CTS) is always on.)	0- no software flow control 1- use software flow control	1	0
WR	8	Write all configurable parameters to non-volatile memory. All settable parameters are stored.	NA	NA	NA
CN	9	Exit AT command mode.	NA	NA	NA
E0	10	No echo in AT command mode.	NA	NA	NA
E1	11	Echo characters in AT command mode.	NA	NA	NA

AT Command	Binary Command #	Description	Parameters	# Bytes Returned	Factory Default
LH	12	Transmit header time	Time in tenths of seconds for the long header. Range: 0 – 0xFF	1	1
FH	13	Force header on next transmit	NA	NA	NA
RE	14	Restore default configuration	NA	NA	NA
ER	15	Set Receive Error Count	Value of error count. This value is reset to 0 after every reset it is not non-volatile	2	0
GD	16	Set Receive Good Count	Value of good count. This value is reset to 0 after every reset it is not non-volatile	2	0
HP	17	Set Network number. Each network uses a different hop sequence. Seven different network numbers are available.	Use this parameter to operate independent networks of 9XStream modules in the same vicinity. Range: 0 – 6H	1	0
MK	18	Set Module Address Mask	Address mask. Only bits set to 1 are used in the address comparison. A global address is an address that has the same bits set as the address mask.	2	FF FF
CC	19	Set command sequence character.	Number for the command character. Range: 20H –7FH	1	2B
VR	20	Firmware version	NA	2	NA
BD	21 V4.08	Set Serial Baud Rate	Number corresponding to Serial Port baud rate. Baud rate doesn't take affect until ATCN command is issued. If the baud command is issued in binary mode it must be stored (ATWR) and new baud rate will take affect after reset. Range: 1 – 6 1-2400 4-19200 2-4800 5-38400 3-9600 6-57600	1	
RT	22 V4.10	RTS/CMD Control	0 – No binary commands accessed with RTS/CMD. 1 – Binary commands are sent when RTS/CMD is asserted.	1	0

AT Command	Binary Command #	Description	Parameters	# Bytes Returned	Factory Default
SY	23 V4.12	Set Sync Timer. This time represents the time that the transmitter and receiver stay in sync after receiving or sending data. Setting this parameter to 20 (0x14) will allow any module to transmit within the next 2 seconds utilizing a header of 8ms instead of 35ms. Use this parameter to reduce communication latency and turn-around time.	Time in tenths of seconds Range: 0 –FFH	1	0

Networking and Addressing

The 9XStream modules utilize three levels of addressing to communicate between modules. This networking hierarchy is depicted in Figure 8 below. Only modules with the matching addresses are able to communicate. The three methods of addressing are: Vendor Identification number, Networks and Module Addresses.

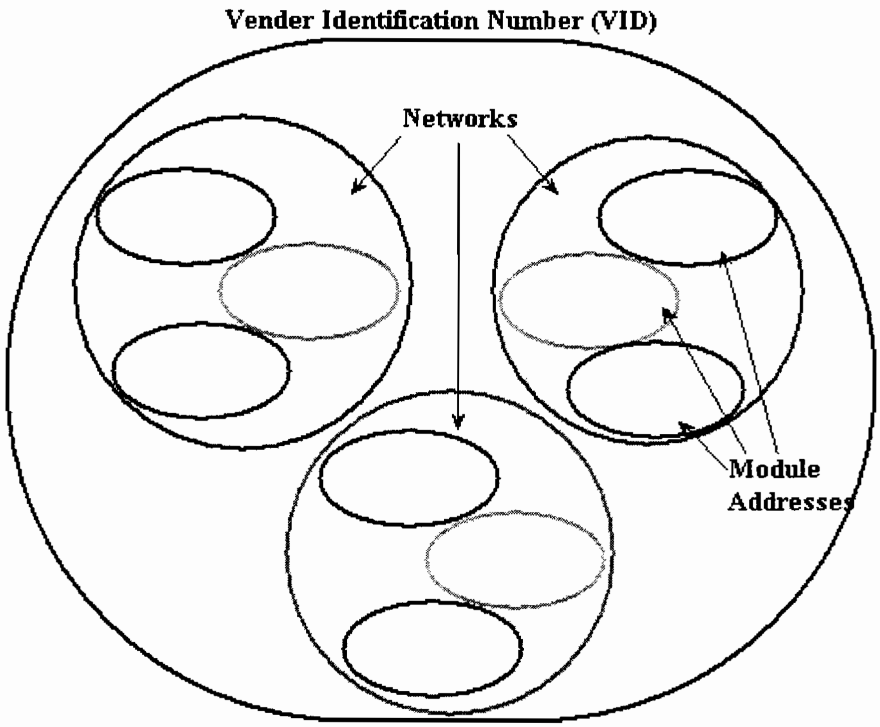


Figure 8 – Layout of a typical network configuration. The 9XStream features a powerful set of networking and addressing options allowing the functionalities of complex networking systems.

Vendor Identification Number (VID)

For network security, a unique Vendor Identification number (VID) can be requested. The VID is programmed into the 9XStream module at the factory and is stored in permanent memory. This number can only be changed at the factory. Only modules with matching VID numbers can communicate together. The VID addressing ensures that modules with a given VID are immune to either transmissions or receptions with 9XStream modules with a different VID in the same vicinity. To request a unique VID number, contact MaxStream to obtain a VID Request Form.

Networks

Within each VID, there are seven available networks. Each network utilizes a different pseudo-random hopping sequence to navigate through the shared hopping channels. In the event that two modules from different networks collide on a channel, because they hop in a different sequence, the two modules will jump to separate channels on the next hop. Using networks, multiple module pairs can operate in the same vicinity with minimal interference from each other. The network parameter is user settable using the "HP" command as described in the "9XStream Command Table" section.

Module Address

Module Addresses and Module Address Masks provide another level of addressing among 9XStream modules. Each module in a network can be configured with a 16-bit Module Address to establish selective communications within a network. This address is set to one of 65535 values using the "DT" command. The default Module Address is 0000H.

All modules with the same Module Address can transmit and receive data among themselves. Any modules on a network with different Module Addresses will still detect and listen to the data in order to maintain network synchronization. However, they will not send the data out to their serial ports if their Module Addresses don't match the Module Address of the transmitter. (The Module Address Mask can be used to provide exceptions to this rule as described in the following section.

Module Address Mask

The Module Address Mask can be used as an additional method to facilitate communication among modules. The Module Address Mask can also be set to one of 65535 possible values using the "MK" command. The default value is FFFFH.

All transmitted data packets contain the Module Address of the transmitting module. When a transmitted packet is received by a module, the Transmitter Module Address (contained in the packet) is logically "ANDed" (bitwise) with the Receiver Module Address Mask. If the resulting value matches the Receiver Module Address, or if it matches the Receiver Module Address Mask, the packet is accepted. Otherwise, the packet is discarded. (When performing this comparison, any "0" values in the Receiver Module Address Mask are treated as "don't care" values and are ignored.)

Packets with a Transmitter Module Address of FFFFH will be received by all modules as shown below in Figure 10. A Transmitter Module Address that matches the Module Address Mask is called a Global Address.

FFFF - Transmitter Module Address
F0F0 - Receiver Module Address Mask

F0F0 - Result from Comparison

Figure 10 – Demonstration of Module Address comparison at receiver. FFFFH logically "ANDed" with any Module Address Mask will equal the Module Address Mask.

The following example helps illustrate the manner in which Module Addresses are checked.

Example

Consider a Module (Module A) with Module Address of 00FFH and Module Address Mask F0F0H as shown in Figure 11 below.

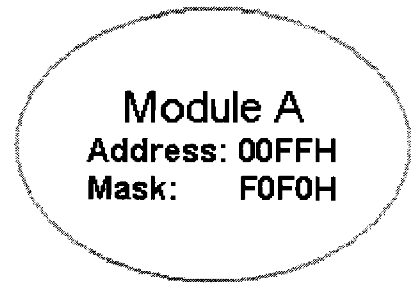


Figure 11 – Diagram of a module with a Module Address of 00FFH and a Module Address Mask of F0F0H.

There are three different ways that Module A can receive packets from other modules.

1) This module could receive packets from other modules with a Transmitter Module Address of 00FFH.

2) Since the Receiver Module Address Mask is set to F0F0H, logically "ANDing" the Mask with the Receiver Module Address yields "0XFX" (HEX) where the 'X' values can be anything. Thus, by setting the Module Address Mask to F0F0H, this module could receive packets from any module with a "0XFX" Module Address.

3) The module can also receive packets from any module with a Module Address that matches the Module Address Mask of the module (F0F0H). However, since the Module Address Mask of Module A (in this example) contains two '0' values, these are don't care values. Hence, Module A could receive packets from any module having a Transmitter Module Address (Global Address) of "FXFX".

Electrical Characteristics

DC Characteristics

Vcc=4.75V to 5.25V

Symbol	Parameter	Condition	Min	Typical	Max	Units
V_{IL}	Input Low Voltage	All input signals	-0.5		0.3*Vcc	V
V_{IH}	Input High Voltage	(Except $\overline{\text{RESET}}$)	0.6*Vcc		Vcc+0.5	V
V_{IH2}	Input High Voltage	($\overline{\text{RESET}}$)	0.9*Vcc		Vcc+0.5	V
V_{OL}	Output Low Voltage	I_{OL}=20mA Vcc=5V			0.6	V
V_{OH}	Output High Voltage	I_{OH}=-3mA Vcc=5V	4.2			V
I_{IL}	Input Leakage Current I/O Pin	Vcc=5V, pin low (abs. value) (Except *CTS, DO, *RESET, *CONFIG)			8.0	uA
I_{IH}	Input Leakage Current I/O Pin	Vcc=5V, pin high (abs. value) (Except RTS/CMD)			980	nA
I_{IL2}		$\overline{\text{CTS}}$, DO, $\overline{\text{RESET}}$		$(Vcc-V_I)/10$		mA
I_{IL3}		$\overline{\text{CONFIG}}$		$(Vcc-V_I)/47$		mA
I_{IH2}		RTS/CMD		$V_I/10$		mA

AC Characteristics

Pin Timings

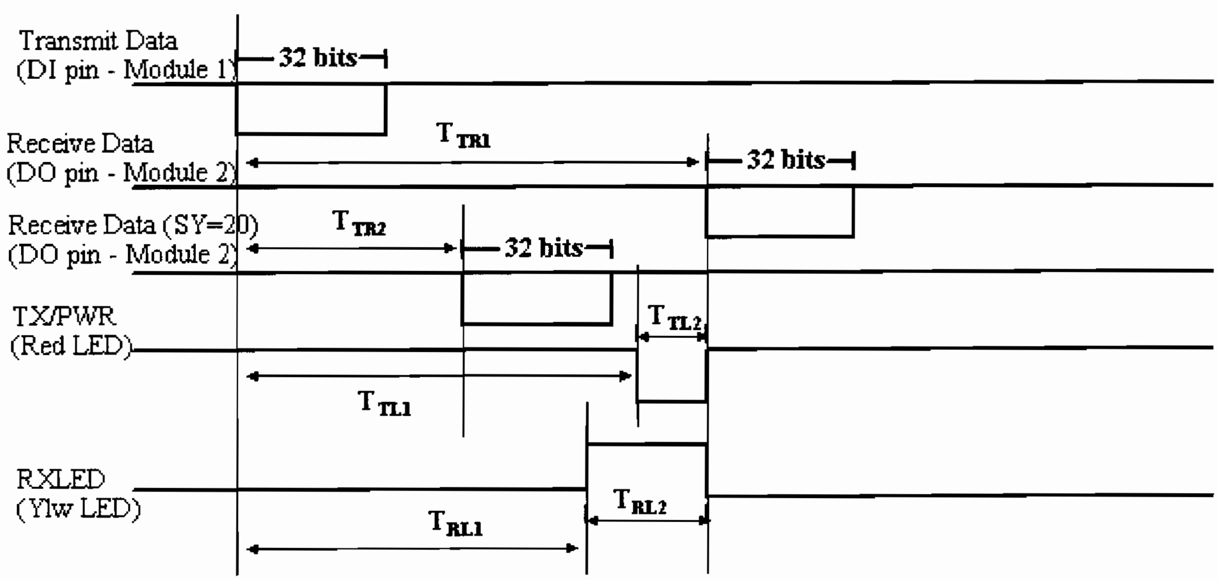
Symbol	Description	009-019XX	009-009XX
T_{R1}	Latency from the time data is transmitted until received.	54 ms	73.6 ms
T_{R2}	Same as T_{R1} (SY=20)	26.6 ms	44.8 ms
T_{TL1}	Time from when data packet is first transmitted until TX/PWR pulses low.	46.4 ms	55.6 ms
T_{TL2}	Time that TX/PWR pin is driven low (when red LED flashes).	8.4 ms	16.8 ms
T_{RL1}	Time from when data packet is first transmitted until RXLED pin goes high on receiver.	40.6 ms	44.8 ms
T_{RL2}	Time that RXLED pin is driven low (when yellow LED flashes).	14 ms	29.6 ms

Cyclic Sleep Mode Timings

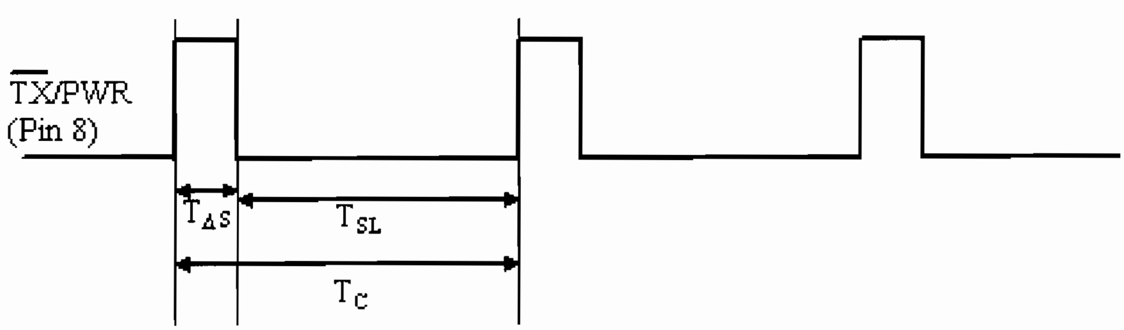
Symbol	Description	19200-baud	9600-baud
$T_{?S}$	Time when module is listening for a valid header to start receiving data.	100 ms	100 ms
T_{SL}	Time where the 9XStream is in its low power cyclic sleep. This time is adjustable using the SM command.	0.5 seconds 1.0 seconds 2.0 seconds 4.0 seconds 8.0 seconds (depends on SM)	0.5 seconds 1.0 seconds 2.0 seconds 4.0 seconds 8.0 seconds (depends on SM)
T_C	Total period between successive pulses of the module awakening from Sleep Mode. This value is dependent on the setting of the SM command.	0.6 seconds 1.1 seconds 2.1 seconds 4.1 seconds 8.1 seconds (depends on SM)	0.6 seconds 1.1 seconds 2.1 seconds 4.1 seconds 8.1 seconds (depends on SM)

Timing Diagrams

Pin Timings



Sleep Mode Timings



FCC Qualifications

IMPORTANT: The 9XSTREAM module has been certified as a module by the FCC for integration into OEM products without further certification being necessary (as per FCC section 2.1091.) The OEM must satisfy the following requirements in order to comply with FCC regulations:

1) The system integrator must ensure that the external label provided with this device is placed on the outside of the final product.

2) In order to comply with the FCC RF exposure requirements, the 9XSTREAM may be used only with Approved Antennas that have been tested with this module and a minimum separation distance of 20 cm must be maintained from the antenna to any nearby persons. The OEM must also include a statement in the final product manual, informing users of the requirement to maintain 20 cm separation from the antenna to any nearby persons. If the OEM integrates the 9XStream into their final product, where the final product utilizes a non-approved antenna or is classified as a portable device per FCC Section 2.1093 (less than 20 cm separation distance between the antenna to any nearby persons), the OEM is responsible for obtaining a separate authorization on the final product.

Warranty

The 9XStream module from MaxStream is warranted against defects in materials and manufacturing for a period of 90 days from the date of purchase. In the event of a product failure due to materials or workmanship, MaxStream will repair or replace the defective product. For warranty service, return the defective product to MaxStream for prompt repair or replacement.

MaxStream, its suppliers, and its licensors shall in no event be liable for any damages arising from the use of or inability to use this product. This includes business interruption, loss of business information, or other losses that may arise from the use of this product.

Application Notes

Why does Sensitivity Matter?

Receiver sensitivity is the lowest power level at which the receiver can detect a signal and demodulate data. Sensitivity is purely a receiver specification and is independent of the transmitter. As the signal propagates away from the transmitter, it attenuates as the distance increases. Lowering the sensitivity on the receiver (making it more negative) will allow the module to detect weaker signals, and can thus increase the transmission range. Sensitivity is vitally important since even slight differences in sensitivity can account for large discrepancies in the range. To better understand this relationship, the following example is provided.

Example:
Compare the MaxStream 9XStream module (with-110 dBm sensitivity) to a commercial radio receiver with a sensitivity of –90 dBm. The Friis transmission formula can be used to calculate received power (or signal strength) at any receiver location under line-of-sight conditions. This formula is given by

$P(r) = P(t)*G(t)*G(r)*(\lambda\hat{}2) / (F(s)*(4*\pi*r)\hat{}2)$, where

$P(r)$ = received power
$P(t)$ = transmitted power
$G(t)$ = gain of transmit antenna
$G(r)$ = gain of receive antenna
$F(s)$ = fading margin
λ = wavelength
r = distance between transmitter and receiver

The following values were used to compare the range limitations of these modules:

$P(t)$=121mW
$G(t)$ and $G(r)$=2dB, or 1.585
λ = 0.333 meters
$F(s)$ = 22 dB, or 158.49 (experimentally determined)

The table below demonstrates the power received at the receiver over the specified range between the TX and RX antennas, assuming line-of-sight conditions.

Range (meters)	Received Power	Detectable by 9XStream module	Detectable by commercial radio
100	-69.177 dBm	YES	YES
500	-83.156 dBm	YES	YES
1000	-89.177 dBm	YES	YES
3000	-98.719 dBm	YES	NO
5000	-103.156 dBm	YES	NO
8000	-107.239 dBm	YES	NO
10000	-109.177 dBm	YES	NO
11265 (7 miles)	-110.21 dBm	YES	NO
12000	-112.699 dBm	NO	NO

Since the range doubles every 6dB, the 20dB sensitivity difference in radios corresponds to $2^{(20/6)} = 10.08$ times the range using the MaxStream radio!

In a similar fashion, MaxStream radios offer 32 times the range of –80 dBm radios and over 100 times the range of –70 dBm radios!

How Does the 'SY' Command Affect Packet Transmission?

Experiment 1 – Byte Transmission

Two 19200-baud 9XStream wireless modules were configured with the sync timer command (SY) set to 20 (2 seconds) and the transmission times were observed. One byte was sent when the modules were out of sync, and was followed (within 2 seconds) by a second byte. Figure A-1 shows the observed results from the oscilloscope. (The pulses have been highlighted in blue and red to highlight the beginning of the transmitted and received data that would otherwise appear faint.)

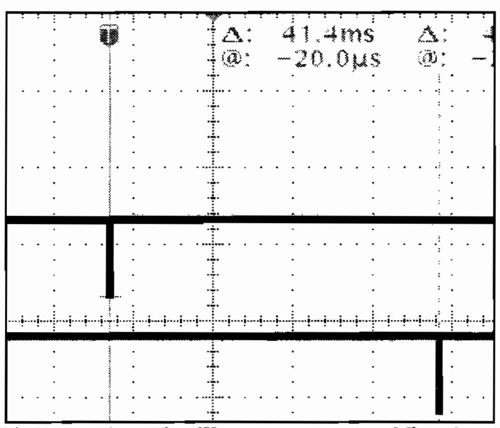


Figure A-1a – Oscilloscope output of first byte on transmit and receive ends.

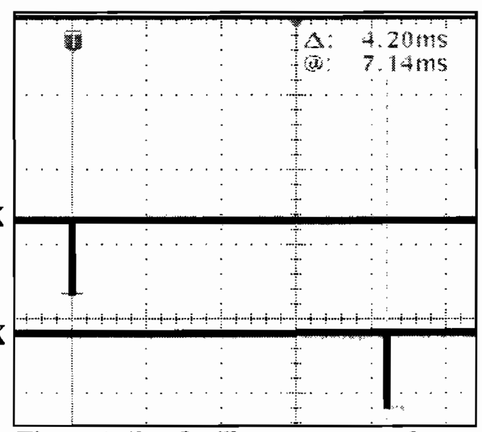


Figure A-1b – Oscilloscope output of second byte with SY set to 20.

From Figure 7 it is evident that the sync timer parameter can save significant amounts of time by reducing the header length. As shown in Figure 7a, the first byte, which included synchronization information in the header, was sent and received in 41.4 ms. Once the modules were synchronized, the second byte transmission did not have the synchronization information included in the header. This transmission occurred in only 4.20 ms – a savings of nearly 90%. Similar testing on the 9600-baud modules showed 48.4 ms to transmit the first byte and synchronize the modules, and 15.8 ms to transmit the second byte without synchronization information.

This experiment was followed-up by a second test to note the effect of the sync timer command on transmitting a continuous data stream.

Experiment 2 – Data Transmission

A continuous stream of 32 byte packets was sent to the 9XStream transmitter through a serial connection and then transmitted to a 9XStream receiver located several feet away. This experiment was performed using 19200-baud modules. The transmission time was first measured with the modules in their default condition, and then measured again after setting the sync timer

(SY command) to 20. The following output plots were obtained from an oscilloscope. (Again, the transmit and receive plots were outlined in blue and red respectively for emphasis.)

TX

RX

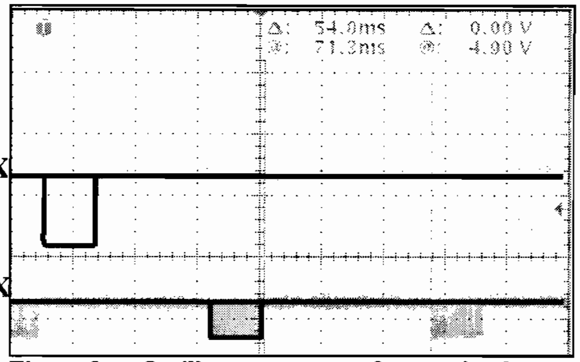


Figure 8a – Oscilloscope output of transmitted and received data under normal conditions.

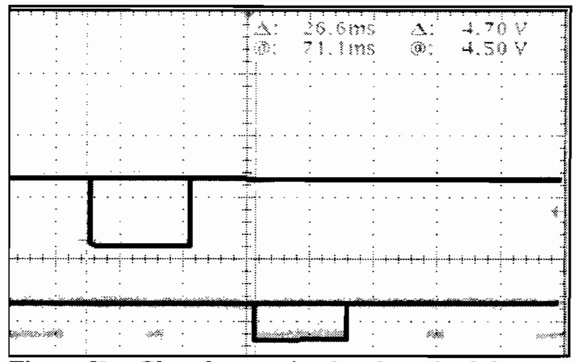


Figure 8b – Plot of transmitted and received data after setting the SY parameter to 20.

In Figure 8a, where synchronization information was transmitted with the data, 54.0 ms was required to transmit each packet to the receiver. After adjusting the SY parameter to stop sending synchronization information in the preamble, Figure 8b shows that the same data transmission occurred in only 26.6 ms.

Using the 9XStream Development Kit

Introduction:

Use this development kit to experience the extended range and ease of use of the MaxStream 9XStream module. This module transmits data at either 9600 or 19200 bits per second (see FCC label on module shield) in the license-free 900MHz ISM band. See how simple it is to communicate with the module using asynchronous serial communications by following the instructions below.

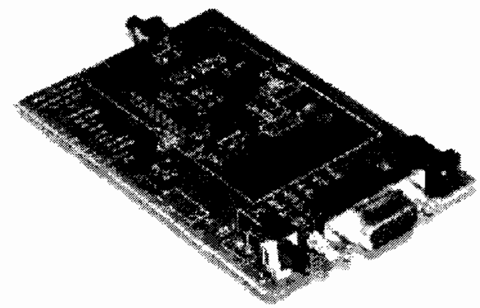


Figure 1 – MaxStream PC Interface Board

Contents:

2	9XStream Frequency Hopping modules
2	Interface Boards
2	9 VDC power supplies
2	DB-9 straight through serial cables
1	Null modem adapter
1	3.5" floppy disk containing file RAT.TXT

Requirements:

2 computers with an available comm port and Windows 95/98 or NT.

Instructions:

1. Assemble and Connect Interface Boards to computers.
   a. Connect the modules to the Interface Boards by fitting the 4 and 11-pin connectors into their respective female headers.
   b. Attach the male sides of one of the DB-9 Connecting Cables to the female DB-9 jacks on the Interface Boards.
   c. Attach the female side of a DB-9 connecting cable to an available RS-232 port on the back of Computer 1. Connect Module 2 to Computer 2 in the same fashion.
   d. Plug power supplies into the Interface Boards. Turn the switch on. The red LEDs on the Interface Boards will illuminate to show the boards are powered.
   e. Refer to figure 1 for completed assembly.

Figure 2 –**Connect To** Dialogue Box

2. Run and configure HyperTrm.exe on both computers
   a. Open the folder on the Start Menu found under Programs -> Accessories -> Communications -> HyperTerminal.
   b. Double-click on the application HyperTrm.exe and choose a name and an icon for the new connection in the **Connection Description** dialogue box.
   c. In the **Connect To** dialogue box on each computer, select *Connect using: Direct to Com1* or *Direct to Com2* to correspond with the comm port used on the respective computers (see figure 2) . Click *OK.*
   d. In the **COM Properties** dialogue box, set the

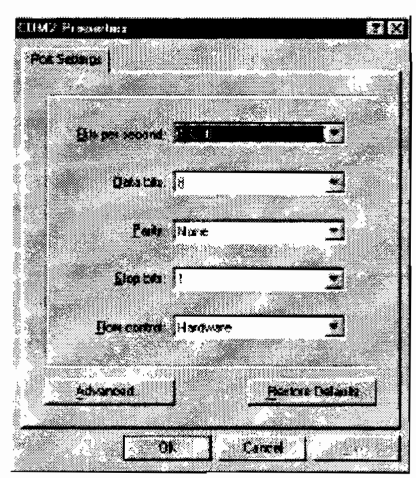


Figure 3 – COM Properties Dialogue Box

following: *Bits per second: 19200; Data bits: 8; Parity: None; Stop Bits: 1* and *Flow Control: Hardware* (See figure 3). Click *OK.* (Set the *Bits per second* to *9600* if you are using 9XStream-96 modules)

e. Both computers are now set up and ready to communicate.

3. Test Connection
   a. Place cursor in HyperTerminal window of Computer 1 and type a message. Notice that text appears only in the window of Computer 2 and text typed on Computer 2 appears only on Computer 1. This text is being transferred over the air by the 9XStream modules.
   b. Transfer the file RAT.TXT
      i. On Computer 1: Select *Transfer* menu -> *Send File*(or try right-clicking in the HyperTerminal window). Insert the disk into the A: drive and enter *Filename: A:\RAT.TXT* Select *Protocol: Ymodem* (See figure 4). Press *Send.*

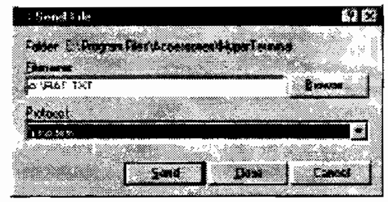


Figure 4 – Send File Dialogue Box

      ii. On Computer 2: Select *Transfer* menu -> *Receive File.* Select a folder in which to place the file and select *Protocol: Ymodem* (see figure 5). Press *Receive.*

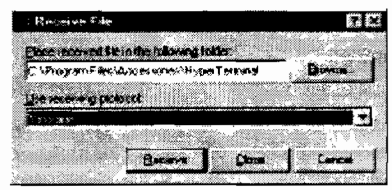


Figure 5 – Receive File Dialogue Box

      iii. The file will begin to transfer and any errors or retries will be accounted for. Touch the antennas and move the modules around to notice that errors or retries do occur under certain circumstances.
      iv. When the transfer completes, open RAT.TXT on Computer 2 and observe the pattern of text, it will be easy to see if any characters were lost or corrupted.
      v. Any file can be transferred this way. Use either Xmodem or Ymodem protocol. Both computers must have the same settings.
   c. Try range testing by distancing the two computers from each other. The modules can send and receive data up to 10 miles line of sight using gain antennas.

4. Problems (Trouble Shooting)
   a. Can't find the HyperTerminal on your computer?
      i. Search your hard drive for HyperTrm.exe
      ii. Download a free version from http://www.hilgraeve.com
   b. Power LED doesn't illuminate when battery clip is attached?
      i. Replace the battery.
   c. Cannot connect to the comm port or communication not working?
      i. Try selecting another comm port by selecting *Properties* under the *File* menu.
      ii. If necessary test the comm ports with a null modem cable (cross RX and TX) between the two computers to verify operation of comm ports.
   d. Characters are getting scrambled?
      i. Verify that both comm ports are set match the baud rate of the modules being tested. Probably 9600 baud or 19200 bits per second. Look for the baud rate on the FCC sticker, 96 or 192 respectively.

Glossary

AT commands – A set of commands that can be used to customize and configure the 9XStream module to meet specific needs. AT commands are sent via a serial communications program such as HyperTerminal.

Binary commands – Another set of commands used to configure the 9XStream module. Binary commands are sent with RTS/CMD asserted. The RT command must be used to enable binary programming prior to using binary commands. Multiple Binary commands can be issued sequentially while RTS/CMD is asserted.

Clear to send – See "CTS pin".

CMOS logic – Logic levels used by the 9XStream module. (0-5V)

Command Mode – A mode of operation wherein the configurable parameters of the 9XStream module can be manually adjusted. Both Binary and AT command modes are available.

Command table – Table containing all 23 currently implemented commands. This table lists all of the adjustable parameters along with a brief description of each.

CRC – See "Cyclic redundancy check".

CTS pin – The low-asserted Clear To Send pin (Pin 1) provides flow control for the 9XStream module. When CTS is asserted (low), serial data can be sent to the module for transmission. If the module is unable to transmit the data, CTS may de-assert (high) once the data buffer nears capacity to prevent buffer overflow.

Cyclic redundancy check (CRC) – Used by the 9XStream module to ensure data integrity during transmission. A CRC is computed on the bits to be transmitted over-the-air and sent with each data packet. The CRC is recomputed by the receiver and compared with the original CRC bits. The packet is valid if the receiver CRC matches the CRC computed by the transmitter.

Cyclic sleep – Sleep Mode setting in which the module enters a low-power state and awakens periodically to determine if any transmissions are being sent.

Data buffer – Collects incoming serial data prior to over-the-air data transmission. The data buffer can hold up to 132 bytes at a given time. When the buffer fills to 115 bytes, the Clear To Send (CTS) pin is de-asserted to stop the flow of incoming serial data.

Data packets – A grouping of data to be sent over-the-air. Each data packet contains a header and data that is collected from the data buffer. The size of the packets varies up to 64 bytes depending on how many bytes of data are in the data buffer.

Data validity – Comparing received data with transmitted data to ensure accurate transmission. Data validity is verified by performing a CRC check.

DI pin – All incoming serial data enters the 9XStream module on the Data In pin (Pin 4).

DO pin – All received over-the-air data leaves the 9XStream module through the Data Out pin (Pin 3) and can then be sent to a microcontroller or RS-232 device.

FCC – The Federal Communications Commission (FCC) is the US government agency responsible for regulating radio communications standards in the US.

Flow control – Method of determining when serial data can be sent to the module for over-the-air transmission. Flow control is used to prevent buffer overflow. This can be implemented in hardware and/or software. Hardware flow control is implemented in the 9XStream module using the *CTS pin.

Frequency Hopping Spread Spectrum (FHSS) – Method employed by the 9XStream module which involves transmitting data over several different channels in a specific channel hopping sequence known by the transmitter and the receiver(s).

Half-duplex – Radios that operate in half-duplex are able to either transmit data or receive data at a given time, but cannot do both simultaneously. The 9XStream is half-duplex. When one module is transmitting, all modules (with the same VID) within range listen to the transmission and will only transmit once the transmission is complete.
Hardware flow control – See "Flow control".
Headers – Information that prefaces the data bits in transmitted data packets. The header contains information used by the receiver(s) to synchronize to the transmitter.
HyperTerminal – A serial communications program useful for communicating with the 9XStream module and configuring user-defined operating parameters through AT commands.
Idle Mode – A mode of operation in which the 9XStream module is neither transmitting nor receiving. In Idle Mode, the module can transition to Receive Mode if over-the-air data is detected, Transmit Mode if serial data is presented on the DI pin, or Command Mode if either Binary or AT command mode is invoked.
Industrial Temperature – Temperature tested version of 9XStream modules extending beyond normal operating specifications (0°C to 70°C). These modules are tested for a temperature range from -40°C to 85°C.
Integration – The simple process of dropping the 9XStream module into an application in place of a serial cable.
Interface board – An optional board available with the 9XStream module that converts RS-232 level data into CMOS logic levels. The interface board is especially useful for developing with the 9XStream module.
Long header – A lengthy header (length determined by LH command) sent out to ensure that modules running in a cyclic sleep mode can detect the header when they awake and synchronize to the transmission.
Low-power modes – See "Sleep Mode".
Module Addresses – Provides a layer of addressing among modules. Modules with the same Module Addresses can communicate together.
Module Address Masks – Provide a layer of filtering to received (over-the-air) data packets. The address (of the transmitting module) is logically "ANDed" with the Module Address Mask of the receiver. The resulting value must match the Module Address of the receiver for the packet to be received. All "0" values are not compared.
Networks – Provides a layer above Module Addresses for communicating between modules. Each network has a unique hopping sequence that allows modules on the same network to remain synchronized to each other.
Pin layout – Describes the layout and functionality of all pins on the 9XStream module.
Pin sleep – This Sleep Mode setting puts the 9XStream into a minimal power state when the SLEEP pin is asserted. It remains in Pin sleep until the SLEEP pin is de-asserted. This setting must be enabled using the SM command.
Power-saving modes – See "Sleep Mode".
Receive Mode – A mode of operation that receives over-the-air data and transmits all valid data packets out to the serial port. The module must be in Idle Mode to transition to Receive Mode.
RS-232 logic – Standard logic levels implemented in all RS-232 devices.
RTS/CMD – The RTS/CMD pin (Pin 5) is used primarily to configure Binary commands (CMD). RTS (Request to Send) flow control is not implemented in the 9XStream module.
Sensitivity – A measurement specification that describes how weak a signal can be (in dBm) and still be detected by the receiver.
Serial data – Data that enters the 9XStream module through its serial port.
Serial port sleep – A Sleep Mode setting wherein the module runs in a low power state until data is detected on the DI pin. This setting must be enabled using the SM command.
Sleep Mode – A mode of operation in which the 9XStream enters a low power-consuming state. Several Sleep Mode settings are available and can be configured using the SM command.

SLEEP pin – If Pin Sleep is enabled, the SLEEP pin (Pin 2) determines if the module is in Sleep Mode or Idle Mode. See "Pin sleep".
Standby Mode – See "Idle Mode".
Start bit – A low UART signal sent to signify the beginning of an eight-bit data sequence.
Stop bit – The last bit in a UART data sequence. The stop bit is high and indicates the end of an eight-bit data sequence.
Synchronization – Synchronization is used to ensure that the transmitter and receiver are in sync with each other and following the same channel hopping sequence.
Transmission Latency – Time required to send a packet of data. This value is dependent on the number of bytes being sent and the baud rate of the module.
Transmit Mode – Mode of operation in which over-the-air data can be transmitted from a module to other modules.
TTL – Transistor-transistor logic.
UART – Universal Asynchronous Receiver-Transmitter. See "Serial port".
VID – The Vendor Identification number (VID) allows modules with the same VID to communicate among themselves. Any module with a different VID will not receive their data transmissions.

Index